Electronic Colloquium on Computational Complexity, Report No. 74 (2004)

On Parallel Pseudorandom Generators (Preliminary version)

Emanuele Viola [*]

August 19, 2004

Abstract

We study pseudorandom generator (PRG) constructions $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ from one-way functions $f : \{0,1\}^n \to \{0,1\}^m$. We consider PRG constructions of the form $G^f(x) = C(f(q_1)\dots f(q_{\text{poly}(n)}))$ where C is a polynomial-size constant depth circuit and C and the q's are generated from x arbitrarily. We show that every black-box PRG construction of this form must jave stretch s bounded as $s \leq l \cdot (\log^{O(1)} n)/m = o(l)$. This holds even if the PRG construction starts from a one-to-one function $f : \{0,1\}^n \to \{0,1\}^m$ where $m \geq 5n$. This shows that either adaptive queries or sequential computation are necessary for black-box PRG constructions with constant factor stretch (i.e. $s = \Omega(l)$) from one-way functions, even if the functions are one-to-one.

On the positive side we show that if there is a one-way function $f : \{0,1\}^n \to \{0,1\}^m$ that is regular (i.e. any $f(x)$ has the same number of preimages) and computable by constant depth circuits then there is a PRG : $\{0,1\}^l \to \{0,1\}^{l+1}$ computable by constant depth circuits. This complements our negative result above because one-to-one functions are regular.

We also study constructions of average-case hard functions starting from worst-case hard ones, i.e. hardness amplifications. We show that if there is an oracle procedure Amp^f in the polynomial time hierarchy (PH) such that Amp^f is average-case hard for every worst-case hard f, then there is an average-case hard function in PH *unconditionally.* Bogdanov and Trevisan (FOCS '03) and Viola (CCC'03) show related negative results under incomparable assumptions. This result is obtained by derandomizing the proof techniques used in our negative result for PRG constructions.

Keywords: Pseudorandom generator, black-box, constant-depth circuits, hardness amplification, restriction

[*] Harvard University. Email: `viola@eecs.harvard.edu` Research supported by NSF grant CCR-0133096 and US-Israel BSF grant 2002246.

ISSN 1433-8092

1 Introduction

A rigorous notion of *pseudorandom generators* (PRGs) was introduced in the seminal works of Blum and Micali [9] and Yao [45] and have since found a striking variety of applications in Cryptography and Complexity Theory. A PRG $G : \{0,1\}^l \to \{0,1\}^{l+s}$ is an efficient procedure that stretches l inputs bits into $l+s$ output bits such that the output of the PRG is indistinguishable from random to efficient adversaries. That is, for every *probabilistic polynomial time machine* (PPT) A we have

$$\Big|\Pr[A(G(U_l)) = 1] - \Pr[A(U_{l+s}) = 1]\Big| \leq \epsilon$$

where U_n denotes a uniform random variable in $\{0,1\}^n$ and ϵ is negligible in $l+s$.

While the existence of PRGs is a major open problem, there has been a series of fascinating works constructing PRGs from weaker and weaker assumptions. Most of these works construct PRGs starting from *one-way functions* [9, 45, 32, 23, 22, 26]. Informally, a function is one-way if it is easy to compute but hard to invert on average. (The existence of one-way functions implies that $P \neq NP$, but the converse is not known to hold.) Yao [45] shows how to construct a PRG from any one-way *permutation*, and this construction is improved [23]. In [22] Goldreich, Krawczyk and Luby show how to construct a PRG from any one-way function, provided that each value in the range of the one-way function has roughly the same number of preimages. Finally, in [26] Håstad et. al. show how to construct a PRG starting from *any* one-way function. For a discussion of pseudorandom generators we refer to the reader to the excellent book by Goldreich [20].

PRG constructions with polynomial stretch: A crucial parameter of every PRG $G : \{0,1\}^l \to \{0,1\}^{l+s}$ is its *stretch* s, that one wants as big as possible. Note that s is only relevant in relation with the input length l, since from a PRG $G' : \{0,1\}^l \to \{0,1\}^{l+1}$ one can trivially construct, for every polynomial p, a PRG $G'' : \{0,1\}^{pl} \to \{0,1\}^{pl+p}$ with stretch p. G'' is the concatenation of the output of p copies of G' on p independent seeds. However, in this way we will never get stretch equal to the seed length (pl).

But in many applications one needs the stretch to be linear in the input length. Two important such applications are Naor's bit-commitment [36] and private-key encryption, where starting from a small key of length l, one wants to generate many bits $l + s \gg l$ that can be used to encrypt messages in a "stream cipher".

So suppose we want to build a PRG $G : \{0,1\}^l \to \{0,1\}^{l+s}$ where $s = l$. To achieve this, using any of the constructions above, one works in two steps. First, starting from a one-way function f one builds a PRG G_1^f with small stretch, say one bit:

$$G_1^f(x) = H^f(x) \circ b^f(x)$$

where $G_1^f(x) : \{0,1\}^l \to \{0,1\}^{l+1}$, $H^f : \{0,1\}^l \to \{0,1\}^l$ and $b^f : \{0,1\}^l \to \{0,1\}$. Then, to get arbitrary polynomial stretch, one uses the following construction due to Goldreich and Micali (see [20], Section 3.3.2): from G_1^f construct $G_2^f : \{0,1\}^l \to \{0,1\}^{l+s}$ defined as

$$G_2^f(x) := b^f(x) \circ b^f(H^f(x)) \circ b^f(H^f(H^f(x))) \circ \cdots \circ b^f(\underbrace{H^f(\cdots(H^f}_{l+s-1}(x)\cdots). \tag{1}$$

Construction (1) is very *sequential* in the following sense: the i-th evaluation of H depends on the output of the $(i-1)$-th evaluation of H, and hence the straightforward circuit for G_2^f has depth at least s.

Notice that, once we have a PRG with linear stretch, say $G : \{0,1\}^l \to \{0,1\}^{2l}$, one can use the more efficient (and parallel) construction of PRGs with arbitrary stretch given by Goldreich, Goldwasser and Micali in [21]. However, there remains the problem of constructing a PRG with linear stretch.

1.1 The main problem we study

The main question addressed in this paper is the following: *Are PRG constructions with arbitrary stretch inherently sequential?* This problem is motivated by the question, both practical and philosophical, of how much cryptography can be done in low complexity classes.

Of course, we must be more precise about what we mean by 'PRG construction' and 'sequential'.

We now discuss PRG constructions. We consider *black-box* PRG constructions, as in many other works starting with the seminal paper by Impagliazzo and Rudich [30]. Roughly speaking, $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ is a *black-box PRG construction from one-way function* f if there is a fixed PPT M such that, for *every* (computationally unbounded) oracle function f and adversary A, if A *breaks* the PRG then M *inverts* f. I.e., if A distinguishes the output of PRG from truly random, then M, when given oracle access to both f and A, can find a preimage of $f(U_n)$ with noticeable probability. The idea is that if $f(U_n)$ cannot be inverted with noticeable probability by a PPT (i.e., if f is one-way) then no PPT can break G^f, and so G^f is a PRG (cf., 2).

Most results in Cryptography, and in particular most PRG constructions (for example [9, 45, 32, 23, 22, 26]) are proved via black-box constructions.

We now define *parallel* PRG constructions. The notion of a parallel PRG construction we look at in this paper is the following:

PRG construction $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ from one-way function $f : \{0,1\}^n \to \{0,1\}^m$

$$G^f(x) = C_x\left(f(q_{x,1}) \dots f(q_{x,\mathrm{poly}(n)})\right)$$

where $C_x : \{0,1\}^{\mathrm{poly}(n)} \to \{0,1\}^{l+s}$ is a constant depth circuit of size $\mathrm{poly}(n)$,
and $C_x, q_{x,1}, \dots, q_{x,\mathrm{poly}(n)}$ are generated from x arbitrarily.

Table 1: Parallel PRG construction.

By a *constant depth circuit* we mean AC_0, i.e. a constant depth circuit with NOT,OR,AND gates, where AND, OR gates have unbounded fan-in (see e.g., [17, 25]).

PRG constructions in the form in Table 1 are intuitively parallel in the following sense: (1) The queries made to f are *non-adaptive* (i.e. they do not depend on f but only on x), and (2) C_x is a *constant depth* circuit, and thus can be evaluated in constant parallel time. It would be interesting to study black-box PRG constructions relaxing either (1) or (2), but we can prove our main results only when both apply.

We do not make any assumption on how $C_x, q_{x,1}, \dots, q_{x,\mathrm{poly}(n)}$ are generated from x: This makes our negative result stronger, while in our positive results all the computation is done by a constant depth circuit.

1.2 Our Results on PRG Constructions

Next we discuss our results for PRG constructions. We have both positive and negative results. We start with the latter.

Theorem 1.1 (This Paper). *Let $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ be a black-box PRG construction (Def. 2.1) in the form in Table* 1*. Then the following hold:*

1. *If G^f starts from one-way function $f : \{0,1\}^n \to \{0,1\}^m$ then $s \leq l \cdot (\log^{O(1)} n)/m$.*
2. *If G^f starts from one-to-one one-way function $f : \{0,1\}^n \to \{0,1\}^{5n}$ then $s \leq l \cdot (\log^{O(1)} n)/5n$.*

Item (1) in Theorem 1.1 shows that black-box PRG constructions from one-way functions can only be parallel if $s = o(l)$ (Typically $m = n^{\Omega(1)}$, so the result gives $s \leq l/n^{\Omega(1)}$). And Item (2) shows this holds even if the PRG construction starts with one-to-one functions, as long as the range of the one-to-one one-way function is sufficiently bigger than its domain.

Note that the bounds in Theorem 1.1 depend on the size of the circuit C_x in Table 1 and on the number of queries made to f *only through* the hidden constant $O(1)$ appearing in the negligible factor $\log^{O(1)} n$.

In Section B we prove that (essentially) Theorem 1.1 Item (1) holds for a less restrictive kind of black-box constructions, i.e. *mildly* black-box constructions.

It seems natural at this point to ask: is there *any* PRG construction in the form in Table 1, even with stretch $s = 1$? The answer is 'yes' if we start from a one-way permutation $\pi : \{0,1\}^n \to \{0,1\}^n$. This is because in this case we can use the following PRG construction given by Goldreich-Levin: $GL^\pi : \{0,1\}^{2n} \to \{0,1\}^{2n+1}$ defined as $GL^\pi(y,r) := (\pi(y), r, \langle y,r\rangle)$, where $\langle y,r\rangle$ is a general hard-core predicate (see [23, 20]). This is clearly of the form in Table 1, since for every input $x = (y,r)$ the circuit C_x that has $\langle y,r\rangle$ and r hardwired and is defined as $C_x(\pi(y)) := \pi(y), r, \langle y,r\rangle$ is trivially constant depth. But what happens if we require that all the computation (in particular $\langle y,r\rangle$) be done in constant depth? It may seem that in this case the answer should be 'no' because now given y, z we must in constant depth compute the general hard-core predicate $\langle y,z\rangle$, but it is known that constant depth circuits cannot compute general hard-core predicates [19].

There is an additional problem for constructions starting from one-way functions f that are not necessarily permutations. Most known PRG constructions (see e.g., [22, 26, 20]) apply pairwise independent hash functions to the evaluations of the one-way functions. In a parallel PRG construction this computation should be done by C_x, which is a constant depth circuit. But, again, it is known that constant depth circuits cannot compute pairwise independent hash functions [35].

Our observation here is that PRG constructions *don't need* to compute general hard-core predicates or pairwise independent hash functions. Indeed, it is sufficient that the circuit computes the output distribution of these objects (e.g., general hard-core predicates) over a random input. And this in fact *can* be done by a constant depth circuit.

Theorem 1.2 (This Paper). *There is a black-box PRG construction $G^f : \{0,1\}^l \to \{0,1\}^{l+1}$ from regular one-way functions $f : \{0,1\}^n \to \{0,1\}^m$ such that G is of the form in Table* 1*. Moreover, all the computation is done by a constant depth circuit of size* $\mathrm{poly}(n)$*. The input length of G is $l = 2n$ starting from one-way permutations $\pi : \{0,1\}^n \to \{0,1\}^n$ and $l = \mathrm{poly}(n)$ starting from regular one-way functions. In particular:*

1. *If there is a one-way permutation $\pi : \{0,1\}^n \to \{0,1\}^n$ computable by constant depth circuits of size* $\mathrm{poly}(n)$ *then there is a PRG $G : \{0,1\}^{2n} \to \{0,1\}^{2n+1}$ computable in constant depth circuits of size* $\mathrm{poly}(n)$*.*
2. *If there is a regular one-way function $f : \{0,1\}^n \to \{0,1\}^m$ computable by constant depth circuits of size* $\mathrm{poly}(n)$ *then there is a PRG $G : \{0,1\}^l \to \{0,1\}^{l+1}$ computable in constant depth circuits of size* $\mathrm{poly}(n)$*.*

A function f is *regular* if any element in the range has roughly the same number of preimages. Note Item (2) in Theorem 1.2 matches Item (2) in Theorem 1.1 because one-to-one functions are regular. Moreover, the ratio s/l of G in Item (1) in Theorem 1.2 matches the bound in Theorem 1.1 up to a polylogarithimic factor. But, G in Item (1) in Theorem 1.2 starts from a permutation, while G in Theorem 1.1 starts with one-to-one functions.

If one allows for $O(\log n)$ bits of non-uniformity in G then we show one can build a PRG in constant depth circuits from any one-way function in constant depth circuits (we omit the details in this preliminary version).

1.3 Related Work

A concurrent and beautiful work related to ours is the one by Applebaum, Ishai and Kushilevitz [5]. They show that the existence of a 'moderately easy' PRG, say in NC_1 (i.e. computable by circuits of polynomial-size and logarithmic-depth), implies the existence of a PRG in NC_0 (i.e. computable by circuits of polynomial-size and constant-depth with *bounded* fan-in). Note that in this work we consider the strictly larger class AC_0 (the class of functions computable by circuits of polynomial-size and constant depth with *unbounded* fan-in.) However, the NC_0 PRG of [5] has *sublinear stretch even if the original NC_1 PRG has polynomial stretch.* This is interesting in relation with our negative results that only rule out parallel black-box PRG constructions with linear stretch. (They also prove analogous connections for other cryptographic primitives, such as one-way functions.) Moreover, they improve on our Theorem 1.2 obtaining the same results for constant-depth circuits with bounded fan-in (whereas our Theorem 1.2 refers to constant-depth circuits with unbounded fan-in, specifically our result uses fan-in $\log^{1+\epsilon} n$).

Another beautiful work related to ours is the one by Gennaro and Trevisan [18]. They show that there is no black-box PRG construction $G^\pi : \{0,1\}^l \to \{0,1\}^{l+s}$ that makes less than $s/\omega(\log n)$ queries to π. This holds even if G^f is a PRG construction from one-way permutations (their work actually rules out a more general kind of black-box construction than the one we consider here, see [41]). The work of Gennaro and Trevisan is thus a tradeoff between the stretch of the PRG construction and the number of queries it makes to f. The difference with our work is the following: We are not concerned with *how many* queries G^f makes to f, rather we are concerned with *how these queries are made and processed.* Again, note that our bounds in Theorem 1.1 essentially do *not* depend on the number of queries made to f (except for the hidden constant in the negligible factor $\log^{O(1)} n$). Rather, they depend on the parallel structure of G.

There exist several other works addressing the complexity of PRGs. Kharitonov, Goldberg and Yung [31] and Yu and Yung [47] prove strong negative results about the ability of various automata and other space-restricted devices to compute PRGs. Linial, Mansour and Nisan [33] prove that constant depth circuits cannot compute pseudorandom functions (an object related to PRGs).

Impagliazzo and Naor [29] show how to construct PRGs based on the assumed intractability of the subset sum problem. In particular, they show how to construct a PRG : $\{0,1\}^n \to \{0,1\}^{n+\log n}$ computable by constant depth circuits. Reif and Tygar [40] and Naor and Reingold [37] construct PRGs under number-theoretic complexity assumptions. In both these works the PRGs constructions are only shown to be computable in circuit classes strictly containing the class we consider. None of the above works addresses the question of constructing a parallel PRG from a *general* assumption, such as the existence of one-way functions. Finally, the complexity of constructing Nisan-Wigderson type PRGs from hard functions is studied in [44]. Some of the proof techniques in this paper (i.e. using the noise sensitivity of constant-depth circuits) are borrowed from [44].

1.4 Worst-case Hardness Amplification

Another problem we study in this paper is the problem of *worst-case hardness amplification*, which is the problem of producing an average-case hard function starting from a worst-case hard function. A motivation for studying this problem is to establish connections between average-case complexity and worst-case complexity, since the latter is much better-understood. This has been accomplished for high complexity classes such *PSPACE* and *EXP* (e.g. [34, 8, 6, 15, 13, 42, 43, 44]). Most constructions in these works are black-box both in the use of the worst-case hard function f and in the 'proof of correctness'. Namely they exhibit efficient algorithms Amp and R such that for *every* function f and *every* adversary A, if A computes Amp^f well on average then R^A computes f everywhere. Note that if f is worst-case hard then R^A cannot be a small circuit. Since R is efficient this means that A cannot be a small circuit, and hence Amp^f is average-case hard.

There are results showing that such connections (i.e., between worst-case and average-case hardness) for classes within the polynomial-time hierarchy (PH) are unlikely to be provable using these kind of black-box techniques: Bogdanov and Trevisan [10], building on [16], show that every hardness amplification within NP such that its proof of correctness is black-box and R is non-adaptive implies that the PH collapses, and therefore such a hardness amplification is unlikely to exist. In a previous work [44] we showed (unconditionally) that there is no hardness amplification within PH where both the use of f and the proof of correctness are black-box.

In this paper we obtain the first negative result on hardness amplifications within PH that are black-box only in the use of f. Specifically, we show that the existence of such hardness amplification procedures is equivalent to the existence of average-case hard functions in PH, in which case no hardness amplification is needed. We give one necessary definition and then our result.

Definition 1.3. *A function $f : \{0,1\}^n \to \{0,1\}$ is* worst-case*-hard (resp., ϵ-hard) for size S if every circuit of size S fails to compute f on some input (resp., on at least ϵ fraction of inputs).*

Theorem 1.4 (This Paper). *Suppose there is a constant a and an oracle machine Amp in PH such that for every $f : \{0,1\}^n \to \{0,1\}$ that is worst-case hard for size $S = S(n)$, $Amp^f : \{0,1\}^{n^a} \to \{0,1\}$ is $.3$-hard for size $S' = S'(n)$. Then there is a constant b and a function f' in PH such that $f' : \{0,1\}^{n^b} \to \{0,1\}$ is $.1$-hard for size S'.*

1.5 Techniques

We now sketch the main ideas in the proof of Theorem 1.1. Similarly to other works [30, 18], the idea is to choose the oracle function f at random from a certain distribution $\tilde{F}$, then show that (1) $\tilde{F}$ is one-way w.h.p. but also (2) there is an adversary that breaks $G^{\tilde{F}}$ w.h.p., thus contradicting the fact that G is a black-box PRG construction. The main new ingredient in this work is that some of the bits in the truth-table of $\tilde{F}$ are *fixed*, and we will give them for free to the adversary. We will then show that for this fixing of bits $G^{\tilde{F}}$ is easy to break. One of the challenges is of course showing that $\tilde{F}$ is still one-way after these bits have been fixed.

More specifically, the bits to be fixed are chosen applying a *random restriction* [17] to the truth table of the oracle f. Since $G^f(x)$ is a constant depth function of evaluations of f, and because after applying a random restriction to a constant depth circuit the circuit 'tremendously simplifies' (see e.g. [17, 25]), it is possible to exhibit an adversary that breaks the output of G. More specifically, we will use the fact that constant depth circuits have *low noise sensitivity* [33, 11, 44], which means that after fixing most of its input bits, a constant depth circuit becomes very biased, i.e. its output does not change much when the few unfixed input bits are filled at random.

We now sketch the main ideas of our negative result about Hardness Amplification, Theorem 1.4. We discuss this result here because it uses the same key ideas used to prove our negative result for black-box PRG constructions, and it also uses some of the derandomization ideas needed to obtain our negative result for *mildly* black-box PRG constructions (Section B). As before, we will choose the oracle function f at random from a certain distribution $\tilde{F}$ where some some of the bits are fixed in such a way that, (1) $\tilde{F}$ is still worst-case hard w.h.p., but (2) $Amp^{\tilde{F}}$ is trivialized. Again, the bits to be fixed are chosen applying a *random restriction* [17] to the truth table of the oracle f. The idea now is that since $Amp^{\tilde{F}}$ is trivialized, we can dispense with the oracle and construct an average-case hard function h from scratch, thus proving the theorem. But the problem is that we don't know what is the fixing of the bits that satisfies (1) and (2). An idea would be to include the fixing of the bits in the input to the function h, but the problem is that the size of this fixing of bits is of the order of the truth table of the oracle f, i.e. 2^n, while we need the input length of h to be polynomial in n (since the circuit size S' in Theorem 1.4 is relative to the input length of f).

To overcome this problem, we *derandomize* the random restriction. I.e., we create a pseudorandom distribution on restrictions that can be generated using only $\text{poly}(n)$ random bits, yet still w.h.p. satisfies (1) and (2). Now, the function h takes σ as part of the input, where σ is of size $\text{poly}(n)$ and is used to generate a pseudorandom restriction. Since now the input length of h is polynomial in n the theorem is proved.

This pseudorandom distribution on restrictions is obtained using Nisan's unconditional PRG against constant depth circuits [38]. The challenges of course are showing that after this derandomization (1) and (2) still hold. In particular, for (2) we show that a constant depth circuit becomes very biased even after applying a pseudorandom restriction.

The idea of using Nisan's generator to derandomize restrictions already appeared in [46].

1.6 Organization

This paper is organized as follows. In Section 2 we discuss notation. In Section 3 we give the proof of our negative result for black-box PRG constructions in the form in Table 1 from one-way function, i.e. we prove Item (1) in Theorem 1.1. The proof of Item (2) in Theorem 1.1 appears in Section 4. The proof of Item (1) in Theorem 1.2 is given in Section 5 (the proofs of the other Items of Theorem 5 are omitted in this preliminary version). The proof of Theorem 1.4 is given in Section 6. We discuss open problems in Section 7. In Appendix B we prove that (essentially) Item (1) in Theorem 1.1 holds for *mildly* black-box PRG constructions as well. Finally, Appendix A contains some proof details.

2 Preliminaries

We denote by U_n the uniform random variable over $\{0,1\}^n$ and by $F : \{0,1\}^n \to \{0,1\}^n$ a uniform random function. $\Delta(x,y)$ is the relative Hamming distance between vectors $x, y \in \{0,1\}^n$, i.e. $\Pr_i[x_i \neq y_i]$. Throughout the paper $\epsilon(n)$ denotes a quantity negligible in n, i.e. $1/n^{\omega(1)}$. We write 'w.h.p.' for 'with high probability', i.e. with probability $1 - o(1)$.

Restrictions: A *restriction* ρ on t bits is an element of $\{0,1,*\}^t$, where we think of the *'s as values yet to be chosen. For $x \in \{0,1\}^t$ we denote by $x|_\rho \in \{0,1\}^t$ the restriction of x with respect to ρ, i.e. the string obtained from ρ by substituting the *'s with the corresponding bits of x. Note $x|_\rho$ only depends on the bits of x corresponding to * in ρ. (In literature, see e.g. [25], restrictions are usually applied to variables, not to bit strings. We find the latter view more convenient here.) We often consider restrictions on bn bits, where b can be as large as 2^n, and it will be convenient to view such restrictions as functions $\rho : [b] \to \{0,1,*\}^n$, where $[b] := \{1,\dots,b\}$. When $\rho : \{0,1\}^n \to \{0,1,*\}^m$ we think of $\rho(i)$ as a partial assignment to the output $f(i)$ of some function $f : \{0,1\}^n \to \{0,1\}^n$. The following is a key definition: for a function $f : \{0,1\}^n \to \{0,1\}^n$ we denote by $f_\rho : \{0,1\}^n \to \{0,1\}^n$ the function defined by

$$f_\rho(x) := f(x)|_{\rho(x)}.$$

For a fixed restriction ρ, a key random variable we will look at is F_ρ (recall $F : \{0,1\}^n \to \{0,1\}^n$ is a uniform random function). Note that F_ρ is the distribution on functions whose truth table is obtained starting from the truth table of ρ and replacing each * with a uniform and independent random bit.

The standard [17] distribution on restrictions R_δ is the one where each symbol in the restriction is independently * with probability δ and otherwise it is a uniform and independent random bit. Note that if $\rho : [b] \to \{0,1,*\}^n$ is random in R_δ then each of the bn symbols in the truth table of ρ is independently * with probability δ and otherwise it is a uniform and independent random bit.

Black-box PRG constructions: Now we formally define black-box PRG constructions.

Definition 2.1 (Black-box PRG Construction). *An oracle machine $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ is a* black-box PRG construction from one-way function $f : \{0,1\}^n \to \{0,1\}^m$ *if there exists an oracle PPT M such that for sufficiently large n, for every $f : \{0,1\}^n \to \{0,1\}^m$ and every $A : \{0,1\}^{l+s} \to \{0,1\}$, if A* breaks G^f, *i.e.*

$$\left|\Pr[A(G^f(U_l)) = 1] - \Pr[A(U_{l+s}) = 1]\right| \geq 1/4$$

then $M^{f,A}$ inverts f, *i.e.*

$$\Pr\left[f(M^{f,A}(f(U_n))) = f(U_n)\right] \geq 1/n.$$

We say G is from one-to-one one-way function *(resp.,* from regular one-way function *if the above is only required to hold when f is one-to-one (resp., regular).*

We think of l, s, m as functions of n. Recall a function $f : \{0,1\}^* \to \{0,1\}$ is regular the number of preimages of $f(x)$ depends only on $|x|$. The values $1/4$ and $1/n$ in Definition 2.1 can be substituted by $1/p(n)$ for any polynomial $p(n)$. We fix them for concreteness. In Definition 2.1, and throughout the paper, probabilities are (implicitly) taken also over the internal coin tosses of the PPTs.

For more on black-box constructions we refer the reader to the survey in the paper by Reingold, Trevisan and Vadhan [41] (in their taxonomy, Definition 2.1 defines a 'fully black-box' PRG construction).

3 Proof of Theorem 1.1

In this section we prove the negative result about black-box PRG constructions, i.e Theorem 1.1. For simplicity we focus on Item (1). We now proceed to sketch the main ideas in the proof.

Suppose $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ is a PRG construction from one-way function $f : \{0,1\}^n \to \{0,1\}^m$, and let M be the inverting machine required by Definition 2.1. The high level idea is to come up with, for sufficiently large n, a function $f : \{0,1\}^n \to \{0,1\}^m$ and a computationally unbounded adversary $A : \{0,1\}^{l+s} \to \{0,1\}$ such that A breaks G^f but $M^{f,A}$ does *not* invert f, thus contradicting the fact that G^f is a PRG construction by Definition 2.1. The construction of f and A will be probabilistic, i.e. we will show a distribution on functions and adversaries that w.h.p. satisfies the above. This certainly ensures that there exist some f, A satisfying the above.

Our final distribution on functions will be F_ρ for a suitable $\rho : \{0,1\}^n \to \{0,1,*\}^m$. Recall from Section 2 that $\rho : \{0,1\}^n \to \{0,1,*\}^m$ is a restriction and that F_ρ is the random function obtained from the truth table of ρ replacing the *'s with random bits.

The main idea in the proof is to find a *fixed* restriction $\rho : \{0,1\}^n \to \{0,1,*\}^m$ that satisfies the following two properties:

I. For every oracle A, with high probability over F, $M^{f,A}$ does not invert F_ρ, i.e.:

$$\Pr_{F,U_n}\left[F_\rho(M^{F_\rho,A}(F_\rho(U_n))) = F_\rho(U_n)\right] \leq \epsilon(n).$$

II. There is a fixed function $g : \{0,1\}^n \to \{0,1\}^n$ such that

$$E_{F,U_l}\left[\Delta\left(G^{F_\rho}(U_l), G^g(U_l)\right)\right] \leq \frac{\operatorname{poly}\log(n)}{m}$$

where E denotes expectation and recall Δ is the relative Hamming distance between the outputs of $G^{F_\rho}(x)$ and $G^g(x)$.

Intuitively, (I) says that F_ρ is hard to invert just because of the randomness left in F even after fixing some of the bits in its truth table according to ρ. (II) says that ρ trivializes $G^{F_\rho}(U_l)$ because, on average, the output of $G^{F_\rho}(U_l)$ is close in Hamming distance to a vector that does not depend on the oracle.

Before discussing how to construct the restriction ρ, let's show how to prove the theorem once we have such a ρ.

Proof sketch Theorem 1.1 Item (1), assuming ρ that satisfies Properties (I) and (II):. We exhibit a (computationally unbounded) adversary which breaks $G^{F_\rho}(U_l)$. Let g be the function given by Property (II), and let d be a sufficiently large constant. Let

$$A_g(z) := 1 \text{ if and only if } \exists x \in \{0,1\}^l : \Delta(G^g(x), z) \leq \frac{\log^d n}{m}.$$

The proof of the theorem then follows from the next claim.

Claim 3.1. *If $s \geq l \cdot \frac{\log^d n}{m}$ then w.h.p. over F, we have that A_g breaks $G^{F_\rho}(U_l)$, i.e.*

$$\left|\Pr_{U_l}[A_g(G^{F_\rho}(U_l)) = 1] - \Pr_{U_{l+s}}[A_g(U_{l+s}) = 1]\right| \geq 1/4.$$

By claim, w.h.p. over F, A_g breaks $G^{F_\rho}(U_l)$, so by definition of black-box PRG construction, M^{F_ρ,A_g} inverts F_ρ w.p. at least $1/n$. This contradicts Property (I).

Proof of Claim 3.1. By Property (II) and Markov's inequality, w.h.p. over F, we have

$$\Pr_{U_l}[A_g(G^{F_\rho}(U_l)) = 1] \geq 1/2.$$

But

$$\Pr[A_g(U_{l+s}) = 1] \leq \frac{|\{z : A_g(z) = 1\}|}{2^{l+s}} \leq \frac{\sum_{x\in\{0,1\}^l} |\{z : \Delta(G^g(x), z) \leq (\log^d n)/m\}|}{2^{l+s}} \leq \frac{2^l \cdot 2^{H(\frac{\log^d n}{m})\cdot(l+s)}}{2^{l+s}}$$

where for every x we bound $|\{z : \Delta(G^g(x), z) \leq (\log^d n)/m\}|$ by $2^{H(\frac{\log^d n}{m})(l+s)}$, where $H(p) = p\log(1/p) + (1-p)\log(1/(1-p))$ is the binary entropy function (see any book on Coding Theory). Since $H(\log^d n/m) \leq \log^{O(1)} n/m$, we have $\Pr[U_{l+s} \in A_g] \leq 1/4$ as

$$s \geq (l+s)\frac{\log^{O(1)} n}{m} + 2.$$

To conclude, note that $l \geq n - \log^2 n$ without loss of generality (otherwise G only queries a negligible fraction of the inputs of f, details omitted). □

□

3.1 Constructing ρ

We now turn to the problem of constructing ρ that satisfies Properties (I) and (II). Again, our construction of ρ will be probabilistic. That is, we will show a distribution on restrictions that satisfies both Properties (I) and (II) w.h.p.. This certainly guarantees the existence of one fixed ρ that satisfies both Properties (I) and (II). We start with some intuition and then we give the actual construction.

Noise Sensitivity of Constant Depth Circuits: For property (II) we use the low noise sensitivity of constant depth circuits. Recall from Section 2 that the standard distribution on restrictions R_δ is the distribution on restrictions where each symbol is independently * with probability δ and otherwise it is a uniform independent random bit.

Lemma 3.2 ([33, 11, 44] Low Noise Sensitivity of Constant Depth Circuits). *Let $C : \{0,1\}^t \to \{0,1\}^{t'}$ be a circuit of size S and depth d. Let $\rho \in R_\delta$ then*

$$E_{\rho \in R_\delta, U_t, U'_t}\Big[\Delta\Big(C(U_t|_\rho), C(U'_t|_\rho)\Big)\Big] \leq O(\delta \cdot \log^{d-1} S).$$

For completeness, we show in Appendix A a simple derivation of Lemma 3.2 from known results [44]. Then, assuming G makes r queries to f, we have the following (taking expectations over random choice of uniform random functions $F, F' : \{0,1\}^n \to \{0,1\}^m$, random input $x \in \{0,1\}^n$ and random $\rho \in R_\delta$):

$$\begin{array}{rcl} E\Big[\Delta\Big(G^{F_\rho}(x), G^{F'_\rho}(x)\Big)\Big] & = & \\ E\Big[\Delta\Big(C_x(F_\rho(q_1), \ldots, F_\rho(q_r)), C_x(F'_\rho(q_1), \ldots, F'_\rho(q_r))\Big)\Big] & = & E\Big[\Delta\Big(C_x(U_{rm}|_\rho), C_x(U'_{rm}|_\rho)\Big)\Big] \quad (2) \\ & \leq & O(\delta \log^{d-1} S) \qquad \text{(By Lemma 3.2)} \end{array}$$

Where Equation (2) follows from the definition of ρ and F_ρ, assuming without loss of generality that $G^{F_\rho}(x)$ never queries the same input twice. So by fixing $F' = g$ and then applying Markov's inequality, we have that most ρ satisfy Property (II) with the expectation at most $O(\delta \log^{d-1} S)$, which is at most $(\operatorname{poly}\log n)/m$, as required by Property (II), when $\delta \leq (\operatorname{poly}\log n)/m$ (recall the size of C_x is $S = \operatorname{poly} n$). The conclusion is that for property (II) the standard distribution R_δ suffices when $\delta \leq (\operatorname{poly}\log n)/m$, and moreover the smaller δ is the better.

F_ρ one-way: But for Property (I) above the opposite is true. This is because we want F_ρ to still be one-way after we fix ρ, even relative to an oracle that depends on ρ (i.e. A). This is intuitively not the case, for example, if we set $\delta = 0$, because now $F_\rho(x) = \rho(x)$ and thus an oracle depending on ρ could help in inverting $F|_\rho(x)$. On the other hand, if $\delta = 1$ then the function is completely random, and as shown in [30] it is one-way w.h.p.. So for Property (I) it seems that the bigger δ is the better. Since for (I) we need $\delta \leq (\operatorname{poly}\log n)/m$, this suggests to choose $\delta := (\operatorname{poly}\log n)/m$.

Our main problem is that the restriction could conceivably leak information about the input. For example, if $m = 2n$, i.e. the range of F is $\{0,1\}^{2n}$, then one could consider the pathological restriction $\rho' : \{0,1\}^n \to \{0,1,*\}^{2n}$ such that for every x the first n symbols of $\rho'(x)$ are x. Now the inversion of $F_\rho(x)$ is trivial, because the output completely reveals the input.

To overcome this problem we need another idea. Intuitively, we need that for most i there is a superpolynomial number of j's such that $\rho(i) = \rho(j)$. This ensures that after we are given $F_\rho(U_n)$ we have little information about U_n, since information-theoretically U_n is uniform on a set of superpolynomial size. To achieve this we compose a random restriction $\rho : [b] \to \{0,1,*\}^n$ with a random function $h : \{0,1\}^n \to [b]$. We now give the formal definition of this distribution and then show that it satisfies properties (I) and (II) w.h.p..

Definition 3.3. *The distribution $\tilde{R}$ on restrictions $\rho_0 \circ h : \{0,1\}^n \to \{0,1,*\}^m$ is defined as follows. Set $\delta := \log^4 n/n$ and $b := n^{\log n}$. Let $\rho_0 : [b] \to \{0,1,*\}^n$ be a random restriction in R_δ. Let $h : \{0,1\}^n \to [b]$ be a random function, then*

$$(\rho_0 \circ h)(x) := \rho_0(h(x))$$

We now show that w.h.p. $\rho_0 \circ h$ satisfies both Properties (I) and (II).

Lemma 3.4. *A random $\rho = \rho_0 \circ h \in \tilde{R}$ satisfies Property (I) w.h.p.*

Proof. We can assume without loss of generality that $(\rho_0 \circ h)(y)$ contains at least $\log^2 n$ *'s for every y. This is because the probability that this does not happen is at most, using union bounds (recall $\delta = \log^4 n/m$, and $b = n^{\log n}$):

$$b\binom{m}{m-\log^2 n}(1-\delta)^{m-\log^2 n} = b\binom{m}{\log^2 n}(1-\delta)^{m-\log^2 n} \leq n^{\log n}(e\cdot m)^{\log^2 n}(1-\delta)^{\delta^{-1}\delta(m-\log^2 n)}$$
$$\leq n^{\log n} e^{O(\log^3 n)}(1/e)^{log^4 n - o(1)} \leq \epsilon(n).$$

Now we fix any such $\rho_0 \circ h$ and we analyze the inversion probability over random F and random input $X \equiv U_n$.

By the pigeon hole principle, there are at most $b2^{n/2}$ inputs x such that there are fewer than $2^{n/2}$ y's such that $h(x) = h(y)$, i.e. $|h^{-1}(h(x))| \leq 2^{n/2}$. Since $b = n^{\log n}$ there is only an exponentially small fraction of such x's. So let's assume without loss of generality that X is such that there are at least $2^{n/2}$ y's such that $h(X) = h(y)$.

In the following we restrict our attention to the queries M makes to the oracle function $F_{\rho_0 \circ h}$, and we make no assumption on the queries it makes to the adversary A. First, without loss of generality assume that M queries its output (clearly it does not hurt for M to check its answer before outputting it) and that M never queries any input twice (this can be accomplished by keeping a simple list of the inputs queried and of the oracle answers).

Since M queries its output, the probability (over F, X) that M inverts $F_{\rho_0 \circ h}(X)$ is the probability that (1) M queries X or (2) M queries $y \neq X$ such that $F_{\rho_0 \circ h}(y) = F_{\rho_0 \circ h}(X)$. We bound these two events separately.

(1): By our assumption, there are at least $2^{n/2}$ y's such that $h(X) = h(y)$. Therefore, given $F_{\rho_0 \circ h}(X)$, X is uniform on a set of inputs of size at least $2^{n/2}$. Hence the probability that M ever queries X is negligible because M only makes $\mathrm{poly}(n)$ queries.

(2): On the other hand, whenever M queries $y \neq X$ then by definition $F_{\rho_0 \circ h}(y) = U'_n|_{\rho_0 \circ h}(y)$, where U'_n is uniform and independent from X and the state of M, since M never queries the same input twice. Since by our assumption $\rho_0 \circ h(y)$ has at least $\log^2 n$ *'s, it follows that with probability at most $1/n^{\log n}$ we have $F_{\rho_0 \circ h}(y) = F_{\rho_0 \circ h}(X)$. Again, since M only makes $\mathrm{poly}(n)$ queries, the probability that M every queries $y \neq X$ such that $F_{\rho_0 \circ h}(y) = F_{\rho_0 \circ h}(X)$ is negligible.

Therefore, the total probability that M inverts $F_{\rho_0 \circ h}(X)$ is negligible. □

Lemma 3.5. *A random $\rho_0 \circ h \in \tilde{R}$ satisfies Property (II) w.h.p.*

Proof. In order to show Property (II) all we need is Equation (2) in Page 9 to go through *approximately.* Let $F, F' : \{0,1\}^n \to \{0,1\}^m$ be random uniform random functions, $x \in \{0,1\}^n$ a random input, $\rho_0 \circ h$ a random restriction in $\tilde{R}$ and ρ a random restriction in R_δ for $\delta := \log^4 n/m$ (i.e. the same δ in Definition 3.3. Consider the random variable

$$V := \Delta\Big(C_x(F_{\rho_0 \circ h}(q_1), \ldots, F_{\rho_0 \circ h}(q_r)), C_x({F'}_{\rho_0 \circ h}(q_1), \ldots, {F'}_{\rho_0 \circ h}(q_r))\Big).$$

We show

$$E\Big[V\Big] \leq E\Big[\Delta\Big(C_x(U_{rm}|_\rho), C_x(U'_{rm}|_\rho)\Big)\Big] + \epsilon(n). \tag{3}$$

Note Lemma 3.5 follows from Inequality 3 as explained before in Page 9: we use Lemma 3.2 to bound $E[\Delta(C_x(U_{rm}|_\rho), C_x(U'_{rm}|_\rho))]$ by $O(\delta \,\mathrm{poly} \log n)$, then we fix $F' = g$ and use Markov's inequality.

So all we need to show is that Inequality 3 holds:

$$E\Big[V\Big] \leq E\Big[V\Big|\forall i \neq j, h(q_i) \neq h(q_j)\Big] + \Pr[\exists i \neq j : h(q_i) = h(q_j)] \leq$$
$$E\Big[\Delta\Big(C(U_{rm}|_\rho), C(U'_{rm}|_\rho)\Big)\Big] + r^2/b$$

Where we use the fact that conditioned on the event "$h(q_i) \neq h(q_j)$ for every $i \neq j$" the induced distribution of $\rho_0 \circ h$ is exactly R_δ. And then we use the fact that a random function mapping in $[b]$ has collision probability $1/b$, i.e., for every $a \neq b$ we have $\Pr_h[h(a) = h(b)] \leq 1/b$. Since G makes only r queries to f, the probability that there are $i \neq j$, such that $h(q_i) = h(q_j)$ is at most r^2/b. Noticing that r^2/b is negligible because $b = n^{\log n}$ and $r = \mathrm{poly}(n)$ concludes the proof. □

4 PRG constructions from one-to-one one way functions

In this section we prove our negative result for black-box parallel PRG constructions from one-to-one one-way functions, i.e. the proof of Item (2) in Theorem 1.1.

The problem is that the functions $F_{\rho_0 \circ h}$ defined in Section 3 are not one-to-one. To ensure this property, we define another distribution on restrictions and from this a new distribution on one-to-one functions. This definition is slightly elaborate because injectivity contrasts with the fact that we need $\rho(x)$ to *not* uniquely identify x (to preserve the one-wayness of the oracle, see Section 3).

We denote by $\rho(x)_k \in \{0, 1, *\}$ the k-th symbol of $\rho(x)$. We say that a restriction $\rho : [b] \to \{0, 1, *\}^{cn}$ *splits* if for every $i \neq k$ there is $k > \log^2 n$ such that $\rho(i)_k = 1$ and $\rho(j)_k = 0$, or $\rho(i)_k = 0$

and $\rho(j)_k = 1$. The idea is that if ρ splits then for every function $f : [b] \to \{0,1\}^{cn}$ we have that the function f_ρ is injective. For technical reasons we require $k > \log^2 n$.

Definition 4.1. *Let $c := 5$. The distribution $\tilde{\overline{R}}$ on restrictions $\overline{\rho_0 \circ h} : \{0,1\}^n \to \{0,1,*\}^{cn}$ is defined in stages as follows. Set $\delta := (\log^4 n)/n$ and $b := 2^{n - \log^2 n}$.*

Let $\overline{h} : \{0,1\}^n \to [b]$ be a random function such that for every $i \leq b$ there are exactly $n^{\log n}$ inputs $x \in \{0,1\}^n$ such that $h(x) = i$.

Let $\rho' : [b] \to \{0,1,*\}^{cn}$ be a random restriction in R_δ such that ρ' splits. (We can think of ρ' as being generated by sampling from R_δ until ρ' splits.)

*Then let the random restriction $\overline{\rho_0}$ be equal to ρ' except for every i if $\rho'(i)$ contains less than $\log^2 n$ *'s, then set $\overline{\rho_0}(i)_j := *$ for every $j \leq \log^2 n$. (I.e. this forces $\overline{\rho_0}(i)$ to have at least $\log^2 n$ *'s for every i.) Then define*

$$\overline{\rho_0 \circ h}(x) := \overline{\rho_0}(\overline{h}(x))$$

By $\overline{F}_{\overline{\rho_0 \circ h}} : \{0,1\}^n \to \{0,1\}^{cn}$ we denote a random one-to-one function consistent with $\overline{\rho_0 \circ h}$, i.e. such that $F(x)|_{\overline{\rho_0 \circ h}(x)} = F(x)$ for every x. In other words, $\overline{F}_{\overline{\rho_0 \circ h}}$ is a random one-to-one function obtained from the truth table of $\overline{\rho_0 \circ h}$ replacing the *'s with random bits, *conditioned on the event that $\overline{F}_{\overline{\rho_0 \circ h}}$ is one-to-one.*

It is easy to check that the space of restrictions $\tilde{\overline{R}}$ is not empty, i.e. there exist restrictions that satisfy Definition 4.1. It is also easy to see that this guarantees that the space of functions $\overline{F}_{\overline{\rho_0 \circ h}}$ is not empty, because in Definition 4.1 ρ splits and, for every $i \leq b$, $\rho(i)$ has at least $\log^2 n$ *'s and finally there are only $n^{\log n}$ inputs mapping to the same i through h.

All that is left to do is to show that $\overline{\rho_0 \circ h}$ satisfies Properties (I) and (II) from Page 8 w.h.p.. Of course, these properties must now be satisfied for our new space of random functions, namely $\overline{F}_{\overline{\rho_0 \circ h}}$.

Lemma 4.2. *A random $\overline{\rho_0 \circ h} \in \tilde{\overline{R}}$ satisfies Property (I) w.h.p.*

Proof. The proof is the same as the proof of Lemma 3.4. Let $X \equiv U_n$ be a random input. As in Lemma 3.4 assume that M queries its output and that never queries the same input twice. First, $\overline{\rho_0 \circ h}(y)$ has at least $\log^2 n$ *'s for every y by construction. Again, given $\overline{F}_{\overline{\rho_0 \circ h}}(X)$, by construction X is uniform over a set superpolynomial size. Since M only makes poly(n) query, M queries X with negligible probability.

Suppose M queries $y \neq X$. We cannot argue anymore that $\overline{F}_{\overline{\rho_0 \circ h}}(y)$ contains at least $\log^2 n$ random bits, because $\overline{F}_{\overline{\rho_0 \circ h}}$ is now one-to-one. However, after i queries made by M, the output of $\overline{F}_{\overline{\rho_0 \circ h}}(y)$ is still uniform on a set of size at least $2^{\log^2 n} - i$. Since M makes only poly(n) queries, $\overline{F}_{\overline{\rho_0 \circ h}}(y)$ is always uniform on a set of size $2^{\log^2 n} - \text{poly}(n) = n^{\omega(1)}$. Therefore the probability that M ever queries $y \neq X$ such that $\overline{F}_{\overline{\rho_0 \circ h}}(y) = \overline{F}_{\overline{\rho_0 \circ h}}(X)$ is negligible.

Therefore, the total probability that M inverts $\overline{F}_{\overline{\rho_0 \circ h}}(X)$ is negligible. □

Lemma 4.3. *A random $\overline{\rho_0 \circ h} \in \tilde{\overline{R}}$ satisfies Property (II) w.h.p.*

Proof. Again, in order to show Property (II) all we need is Inequality (2) in Page 9 to go through approximately. Let $x \in \{0,1\}^l$ be a random input, $q_1, \ldots, q_r$ the $r \leq \text{poly}(n)$ queries made by C_x, $\overline{\rho_0 \circ h}$ be random in $\tilde{\overline{R}}$, and let ρ simply be a random restriction in R_δ (for $\delta := (\log^4 n)/n$ as in Definition 4.1).

Consider the random variable

$$V := \Delta\Big(C_x(\overline{F}_{\overline{\rho_0 \circ h}}(q_1), \ldots, \overline{F}_{\overline{\rho_0 \circ h}}(q_r)), C_x(\overline{F'}_{\overline{\rho_0 \circ h}}(q_1), \ldots, \overline{F'}_{\overline{\rho_0 \circ h}}(q_r))\Big).$$

As in Lemma 3.5, Lemma 4.3 follows from the following inequality:

$$E\Big[V\Big] \le E\Big[\Delta\Big(C(U_{rn}|_{\overline{\rho_0 \circ h}}), C(U'_{rn}|_{\overline{\rho_0 \circ h}})\Big)\Big] + \epsilon(n). \tag{4}$$

We now prove Inequality 4.

$$\begin{aligned}
E\Big[V\Big] &\le E\Big[V\Big|\forall i \neq j : h(q_i) \neq h(q_j)\Big] + \Pr\Big[\forall i \neq j : h(q_i) \neq h(q_j)\Big] \\
&\le E\Big[V\Big|\forall i \neq j : h(q_i) \neq h(q_j)\Big] + \epsilon(n) \\
&= E\Big[\Delta\Big(C(U_{rn}|_{\overline{\rho_0}}), C(U'_{rn}|_{\overline{\rho_0}})\Big)\Big] + \epsilon(n) \\
&\le E\Big[\Delta\Big(C(U_{rn}|_{\rho}), C(U'_{rn}|_{\rho})\Big)\Big] + \epsilon(n)
\end{aligned}$$

Where the second inequality follows from the fact that $b = n^{\omega(1)}$ (details omitted). And the last inequality follows from the fact that any $r = \mathrm{poly}(n)$ fixed values of $\overline{\rho_0}$ look like the corresponding values of ρ w.h.p.. To show this latter claim we need to bound two probabilities.

First, the probability that $\rho \in R_\delta$ does not split can be bound as follows. Fix $i \neq j$. The probability that does not exist $k > \log^2 n$ such that $\rho(i)_k = 1$ and $\rho(i)_k = 0$, or $\rho(i)_k = 0$ and $\rho(i)_k = 1$ is at most

$$(\delta + \delta + 1/2)^{cn - \log^2 n} \le (2/3)^{4n}.$$

So by a union bound the probability that there are $i \neq j$ such that does not exist $k > \log^2 n$ such that $\rho(i)_k = 1$ and $\rho(i)_k = 0$, or $\rho(i)_k = 0$ and $\rho(i)_k = 1$ is at most

$$(2^n)^2 \cdot (2/3)^{4n}$$

which is negligible.

Second, the probability that there exists i such that $\rho(i)$ has less than $\log^2 n$ *'s is at most

$$r\binom{cn}{n - \log^2 n}(1-\delta)^{cn - \log^2 n}$$

which is negligible (cf. the similar bound inside the proof of Lemma 3.4). □

5 PRG constructions in constant depth circuits

In this section we sketch the proof of Theorem 1.2. We give the details of Item (1) only, which we now state.

Theorem 5.1. *If there is a one-way permutation $f : \{0,1\}^n \to \{0,1\}^n$ in constant depth circuits then there is PRG $G : \{0,1\}^{2n} \to \{0,1\}^{2n+1}$ in constant depth circuits.*

We use the same pseudorandom distribution of Goldreich-Levin [23], and our only difficulty is showing how it can be generated in constant depth circuits. We denote by $\langle x, y\rangle$ the Goldreich-Levin general hard-core predicate [23], i.e. $\sum_i x_i y_i \pmod 2$.

Theorem 5.2 ([23]). *Let $f : \{0,1\}^n \to \{0,1\}^n$ be a one-way permutation. Then $GL^f(x,y) := (f(x), y, \langle x, y\rangle)$ is a PRG.*

At first glance GL^f does not seem to be computable in constant depth circuits, because parity is not [17, 25]. In the following lemma we show how to circumvent this problem.

Lemma 5.3. *There is a constant depth circuit* $C : \{0,1\}^{2n} \rightarrow \{0,1\}^{2n+1}$ *such that for every* $x \in \{0,1\}^n, C(x, U_n)$ *is distributed as* $(U'_n, \langle x, U'_n \rangle)$.

Theorem 5.1 follows from Lemma 5.3 simply defining $G^f(x,y) := (f(x), C(x,y))$.

The key observation to prove Lemma 5.3 is that while constant depth circuits cannot compute the parity function, constant depth circuits *can* generate a random x together with its parity. To see this, consider the constant depth circuit $C : \{0,1\}^n \rightarrow \{0,1\}^{n+1}$ such that $C(r_1, \ldots, r_n) := (r_1, r_2 \oplus r_1, r_3 \oplus r_2, \ldots, r_n \oplus r_{n-1}, r_n)$. It is easy to see that $C(U_n)$ outputs a random value in $\{0,1\}^n$ and its parity, and moreover C is constant depth. This observation is from [12].

To prove Lemma 5.3 we use the same approach, but only on the bits of x that are 1.

Proof of Lemma 5.3. Let the input be $x_1, \ldots, x_n, r_1, \ldots, r_n$ and consider the circuit $C : \{0,1\}^{2n} \rightarrow \{0,1\}^{n+1}, C(x,r) = r'b$, where $r' = r'_1, \ldots, r'_n$ and $b \in \{0,1\}$, defined as follows:

$$r'_i := \begin{cases} r_i & \text{if } x_i = 0 \\ r_i & \text{if } x_i = 1 \text{ and there is no } j < i : x_j = 1 \\ r_i \oplus r_j & \text{if } x_i = 1 \text{ and } j \text{ is the biggest index} j < i : x_j = 1 \end{cases}$$

and

$$b := r_i \text{ where } i \text{ is the biggest index such that } x_i = 1 \qquad (b = 0 \text{ if there is no such } i).$$

It is easy to see that $C(x, U_n)$ is distributed like $(U_n, \langle x, U_n \rangle)$. It is also easy to check that C can be implemented in constant depth. (Indeed, this follows from the fact that we defined it using first-order logic, see [7]). □

For constructions from one-to-one and regular one-way functions the only additional thing we need are *extractors.* While constant-depth circuits cannot in general compute extractors with good parameters ([44]), it can be shown that they can compute extractors (specifically, one based on the hash function due to Carter and Wegman [14]) for the parameters of interest here (i.e. seed length polynomial in the source length) (details omitted).

6 Worst-case Hardness Amplification

In this section we prove Theorem 1.4. Before proving it we make some remarks.

Remarks on Theorem 1.4:

- We focus on amplification up to constant (i.e. .3). Notice that by Yao's XOR lemma (cf., [24]) if PH has a $1/\operatorname{poly}(n)$-hard function for size $S'(n)$ then PH has a .3-hard function for size $S'(n^\Omega(1))$.
- We assume without loss of generality $S \leq 2^n/n^{\omega(1)}$ because for $S \geq 2^n/n^c$ for some fixed constant c, the oracle is already $1/\operatorname{poly}(n)$-hard by a counting argument (given in [44]), and therefore by the previous item PH has a .3-hard function.
- Similar results hold for hardness amplifications $Amp^f : \{0,1\}^{l(n)} \rightarrow \{0,1\}$ running in time $t(n)$ with a constant number of alternations. Here we set $l(n) = \operatorname{poly}(n)$ and $t(n) = \operatorname{poly}(n)$ for simplicity of exposition.

In this section functions are boolean. In particular F will denote a uniform random function $F : \{0,1\}^n \to \{0,1\}$. Accordingly, we take restrictions ρ on 2^n bits, $\rho : \{0,1\}^n \to \{0,1\}$, which we see as a partial assignment to the truth table of $f : \{0,1\}^n \to \{0,1\}$.

To prove Theorem 1.4 we build a certain pseudorandom distribution on restrictions. This is the main technical lemma of this section.

Lemma 6.1. *For every constant c there is a distribution $\tilde{R}^c$ on restrictions $\rho_\sigma : \{0,1\}^n \to \{0,1,*\}$ such that*

1. *There is a polynomial time algorithm such that given $poly(n)$ random bits σ and $x \in \{0,1\}^n$, generates a random $\rho_\sigma \in \tilde{R}^c$ and returns $\rho_\sigma(x) \in \{0,1,*\}$.*
2. *For every circuit C of size 2^{n^c} and depth c on $t := 2^n$ bits: w.p. $1 - o(1)$ over $\rho_\sigma \in \tilde{R}^c$,*
$$\mathrm{Bias}_{U_t}[C(U_t|_{\rho_\sigma})] \geq 1 - o(1)$$
(Where the Bias *of a $0-1$ random variable X is $|\Pr[X = 0] - \Pr[X = 1]|$).*
3. *W.h.p. over $\rho_\sigma \in \tilde{R}^c$, ρ_σ has at least $2^n/n^{O(1)}$ $*$'s.*

We now assume the above Lemma and prove Theorem 1.4.

Proof of Theorem 1.4. By standard techniques (see e.g., [17, 25]), the oracle algorithm in PH can be turned into an exponential size constant-depth circuit whose input is the truth table of the oracle f. In particular, let c be such that $Amp^f(x)$ has depth c and size 2^{n^c} when turned into a constant depth circuit whose only input is the truth table of f (see [17, 25]). Consider the distribution on restrictions $\tilde{R}^c$ whose existence is guaranteed by Lemma 6.1. Consider $Amp^{F_{\rho_\sigma}}$, where $\rho_\sigma \in \tilde{R}^c$. We need a couple of lemmas.

Lemma 6.2. *W.h.p. over $\rho_\sigma \in \tilde{R}^c$ and F, $F_{\rho_\sigma} : \{0,1\}^n \to \{0,1\}$ is worst-case hard for size $S(n)$. In particular, w.h.p. over $\rho_\sigma \in \tilde{R}^c$ and F, $Amp^{F_{\rho_\sigma}} : \{0,1\}^{n^a} \to \{0,1\}$ is .3-hard for size $S'(n^a)$.*

Lemma 6.3. *There is a PH machine A' such that given σ and an input x rounds $Amp^{F_{\rho_\sigma}}(x)$ to its most likely value, over the choice of F, whenever $\mathrm{Bias}_F[Amp^{F_{\rho_\sigma}}(x)] \geq 1 - o(1)$. I.e., if $\Pr_F[Amp^{F_{\rho_\sigma}}(x) = 1] \geq 1 - o(1)$ then $A'(\sigma, x) = 1$, and if $\Pr_F[Amp^{F_{\rho_\sigma}}(x) = 0] \geq 1 - o(1)$ then $A'(\sigma, x) = 0$.*

Now for the proof of Theorem 1.4. By Lemma 6.1, for every x, w.p. $1 - o(1)$ over σ, $\mathrm{Bias}_F[Amp^{F_{\rho_\sigma}}(x)] \geq 1 - o(1)$. This holds because, for fixed x, $Amp^f(x)$ is a constant depth function of the truth table of f. Let A' be the oracle PH machine in Lemma 6.3. By Lemma 6.3 we have:

$$\Pr_{x,\sigma,F}[A'(\sigma, x) \neq Amp^{F_{\rho_\sigma}}(x)] \leq o(1) + o(1) = o(1). \tag{5}$$

Thus

$$\Pr_{\sigma,F}[\Delta(A'(\sigma, .), Amp^{F_{\rho_\sigma}}) \geq o(1)] \leq o(1). \tag{6}$$

Where $\Delta(f, f')$ denotes the relative Hamming distance of the truth tables of $f, f' : \{0,1\}^n \to \{0,1\}$. Thus:

$$\begin{aligned}
&\Pr_{\sigma,F}[A'(\sigma,.)\text{is not .2-hard for size} S'(n^a)] \leq \\
&\leq \Pr_{\sigma,F}[\Delta(A'(\sigma,.), Amp^{F_{\rho_\sigma}}) > .1 \text{ or } Amp^{F_{\rho_\sigma}} \text{ is not .3-hard for size} S'(n^a)] \\
&\leq o(1) + o(1) \qquad \text{(By Inequality (6) and Lemma 6.2)} \\
&\leq o(1).
\end{aligned}$$

(where we use the fact that if $A'(\sigma, .)$ is .1 close in relative Hamming distance to a $Amp^{F_{\rho_\sigma}}$ that is .3-hard, then $A'(\sigma, .)$ must be .2-hard. For else the same circuit computing $A'(\sigma, .)$ on more than $1 - .2$ fraction of inputs would compute $Amp^{F_{\rho_\sigma}}$ with error less $.2 + .1 = .3$).

Now, since w.h.p. over σ we have that $A'(\sigma, .)$ is .2-hard for size $S'(n^a)$, we have that every circuit of size $S'(n^a)$ fails to compute A' on at least a $.2(1 - o(1)) > .1$ fraction of inputs, and thus A' is .1-hard for size $S'(n^a)$.

To finish the proof note that A' is in PH and that it has input length n^b for some b. □

Proof of Lemma 6.3. This lemma was essentially proved by Nisan in [38] (see also [39]) (In [38, 39] the lemma is stated as "almost-PH=PH"). We omit the details here. □

Note Lemma 6.3] does not immediately follow from the more well-known fact that $BPP \subseteq PH$ (even though the latter result is used in Nisan's proof). This is because in 6.3 the machine Amp has access (through the oracle) to an exponential (as opposed to polynomial) number of random bits.

6.1 Pseudorandom restrictions

In this section we prove Lemma 6.1. A key tool is Nisan's pseudorandom generator against constant depth circuits.

Theorem 6.4 ([38]). *For every c there is $Nis : \{0,1\}^{\text{poly}\log n} \to \{0,1\}^n$ such that (1) given x and $i \leq n$ we can compute the i-th bit of $Nis(x)$ in time* $\text{poly}\log(n)$ *and (2) N is* $1/n$-pseudorandom *for circuits of size n and depth c. That is, for every circuit $C : \{0,1\}^n \to \{0,1\}$ of size n and depth c:*

$$\left|\Pr[C(Nis(U_{\text{poly}\log n})) = 1] - \Pr[C(U_n) = 1]\right| = 1/n.$$

Proof of Lemma 6.1. Let $\delta := 1/n^{c^2}$. We know by Lemma 3.2 that for every circuit $C : \{0,1\}^t \to \{0,1\}$ of size 2^{n^c} and depth c on t bits:

$$\Pr_{\rho\in R_\delta, U_t, U_t'}[C(U_t|_\rho) \neq C(U_t'|_\rho)] \leq O(\delta log^{c-1} 2^{n^c}) = o(1). \tag{7}$$

The above equation in turn implies that w.p. $1 - o(1)$ over R_δ, $\text{Bias}_{U_t}[C(U_t|_\rho)] \geq 1 - o(1)$ (see below). Moreover by a Chernoff bound the fraction of $*$'s in $\rho \in R_\delta$ will be concentrated around δ.

So R_δ satisfies Items (2) and (3) in Lemma 6.1. But the problem is that R_δ requires at least 2^n bits to be generated, while we aim to a distribution on restrictions which can be generated with $\text{poly}\, n$ bits. To this aim we *derandomize* R_δ.

Let W be a canonical circuit that given $I := 2^n(\log(1/\delta + 1))$ random bits generates R_δ. It is easy to see that there is such a circuit W of size $\text{poly}(2^n)$ and depth $O(1)$.

We now define $\tilde{R}^c$. Consider $N : \{0,1\}^{\log^d I} \rightarrow \{0,1\}^I$ which is $2^{n^{c'}}$-pseudorandom against depth c', for a constant c' to be determined later. For every c' there is a constant d such that such an N exists according to theorem 6.4. Then

$$\tilde{R}^c := W(Nis(U_{\log^d I})).$$

We now prove that $\tilde{R}^c$ has the required properties.

1. By construction $\tilde{R}^c$ can be generated with poly(n) random bits. As shown in [38], given i and the random bits σ we can compute the i-th symbol of $\rho_\sigma \in \tilde{R}^c$ in polynomial time.

2. Let $t := 2^n$. First we show that, under such restrictions, circuits of size 2^{n^c} and depth c still have low noise sensitivity, the claim about the bias then easily follows. To show this we use an approach similar to one used in [27]. We notice that the noise sensitivity of a constant depth circuit C equals the acceptance probability of another (slightly bigger) constant depth circuit C'. Given a restriction ρ, C' tosses coins for U_t and U'_t and answers 1 if and only if $C(U_t|_\rho) \neq C(U'_t|_\rho)$. It is easy to see that such a C' can be implemented in constant depth. Combining C' with our constant depth circuit W that given random bits generates a random restriction in R_δ we obtain another constant depth circuit $C'' := C' \circ W$. Now, the acceptance probability of C'' over a truly random input is the noise sensitivity of C with respect to R_δ, while the acceptance probability of C'' over a pseudorandom input generated using Nisan's PRG is the noise sensitivity of C with respect to $\tilde{R}^c$. Therefore, since C'' cannot distinguish the output of Nisan's PRG from truly random, we deduce that the noise sensitivity of C with respect to $\tilde{R}^c$ is close to the noise sensitivity of C with respect to R_δ.

   Therefore, for every $C : \{0,1\}^t \rightarrow \{0,1\}$ of size 2^{n^c} and depth c:

$$\begin{aligned}\Pr_{\rho_\sigma \in \tilde{R}^c, U_t, U'_t}[C(U_t|_{\rho_\sigma}) \neq C(U'_t|_{\rho_\sigma})] &\leq \Pr_{\rho \in R_\delta, U_t, U'_t}[C(U_t|_\rho) \neq C(U'_t|_\rho)] + o(1) \qquad \text{(by pseudorandomness)} \\ &\leq o(1) \qquad \text{(by Equation (7))}\end{aligned}$$

   In particular,

$$\Pr_{\rho_\sigma \in \tilde{R}^c}\left[\Pr_{U_t, U'_t}[C(U_t|_{\rho_\sigma}) \neq C(U'_t|_{\rho_\sigma})] \leq o(1)\right] \geq 1 - o(1).$$

   Noticing that $\Pr_{U_t, U'_t}[C(U_t|_{\rho_\sigma}) \neq C(U'_t|_{\rho_\sigma})] \leq o(1)$ implies that $\text{Bias}_{U_t}[C(U_t|_{\rho_\sigma})] = 1 - o(1)$ concludes the proof of this item.

3. Ajtai [2] shows the following (see also [3])

   Lemma 6.5 ([2]). *For every i there is a circuit C of size* poly 2^n *and depth $O(1)$ such that, given t and a bit string of length 2^n:*

   - *If the bit string has more than $t + 2^n/n^i$ occurrences of '1' then C outputs* 1.
   - *If the bit string has fewer than $t - 2^n/n^i$ occurrences of '1' then C outputs* 0.

   We expect R_δ to have $\delta 2^n$ $*$'s. By a concentration bound the probability that it has less than $(\delta 2^n)/2$ is $o(1)$. Since $\delta = 1/n^{c^2} = 1/\operatorname{poly} n$, by Lemma 6.5 there is a constant depth circuit of size $poly(2^n)$ that can distinguish the cases, say, "more than $\delta/2$ fraction of $*$'s" and "less than $\delta/3$ fraction of $*$'s" (setting $t = (\delta 2^n)/2.5$ in Lemma 6.5). Therefore, choosing a sufficiently large constant c' in the definition of $\tilde{R}^c$, by pseudorandomness, we have that $\rho_\sigma \in \tilde{R}^c$ has at least $(\delta 2^n)/3$ $*$'s w.h.p..

□

7 Open Problems

- Is there a parallel black-box PRG construction with linear stretch from one-way permutations?
- Is there a *uniform* parallel black-box PRG construction with any stretch from one-way functions? Our techniques only give a nonuniform one.
- In Theorem 1.2, Item (2), can we achieve $l = (n+m)\,\mathrm{poly}\log n$? This would match Theorem 1.1 Item (2). It would be enough to show the existence of an extractor (that extracts almost all the min-entropy) with linear seed length and computable by constant depth circuits. Such an extractor is not ruled out by [44] nor given by our positive results, as we only get polynomial seed length.

8 Acknowledgements

Many thanks to Salil Vadhan for helpful discussions, pointing out a mistake in an early version of this work and for his helpful reading of this paper. We thank Oded Goldreich for helpful discussions and comments. We thank Alex Healy for rediscovering with us some of the results in [12] that then led to Lemma 5.3, and for his helpful reading of this paper. We thank Yuval Ishai for sending us an early version of [5] and for helpful discussions about PRGs. We thank Minh Nguyen for her helpful reading of the introduction.

References

[1] *Proceedings of the Twenty First Annual ACM Symposium on Theory of Computing*, Seattle, Washington, 15–17 May 1989.

[2] Miklós Ajtai. Σ_1^1-formulae on finite structures. *Ann. Pure Appl. Logic*, 24(1):1–48, 1983.

[3] Miklós Ajtai. Approximate counting with uniform constant-depth circuits. In *Advances in computational complexity theory (New Brunswick, NJ, 1990)*, pages 1–20. Amer. Math. Soc., Providence, RI, 1993.

[4] Miklós Ajtai and Michael Ben-Or. A theorem on probabilistic constant depth computation. In ACM, editor, *Proceedings of the sixteenth annual ACM Symposium on Theory of Computing, Washington, DC, April 30–May 2, 1984*, pages 471–474, 1984. ACM order no. 508840.

[5] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography in NC^0. In *45th Annual Symposium on Foundations of Computer Science*, Rome, ITALY, 17–19 October 2004. IEEE.

[6] László Babai, Lance Fortnow, and Carsten Lund. Nondeterministic exponential time has two-prover interactive protocols. *Comput. Complexity*, 1(1):3–40, 1991.

[7] David A. Mix Barrington, Neil Immerman, and Howard Straubing. On uniformity within NC^1. *J. Comput. System Sci.*, 41(3):274–306, 1990.

[8] Donald Beaver and Joan Feigenbaum. Hiding instances in multioracle queries. In *7th Annual Symposium on Theoretical Aspects of Computer Science*, volume 415 of *Lecture Notes in Computer Science*, pages 37–48, Rouen, France, 22–24 February 1990. Springer.

[9] Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudorandom bits. *SIAM J. on Computing*, 13(4):850–864, November 1984.

[10] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for NP problems. In *44th Annual Symposium on Foundations of Computer Science*, Cambridge, Massachusetts, 11–14 October 2003. IEEE.

[11] Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Inform. Process. Lett.*, 63(5):257–261, 1997.

[12] Ravi B. Boppana and J. C. Lagarias. One-way functions and circuit complexity. *Inform. and Comput.*, 74(3):226–240, 1987.

[13] Jin-Yi Cai, A. Pavan, and D. Sivakumar. On the hardness of the permanent. In *16th International Symposium on Theoretical Aspects of Computer Science*, Lecture Notes in Computer Science, Trier, Germany, March 4–6 1999. Springer-Verlag. To appear.

[14] J. Lawrence Carter and Mark N. Wegman. Universal classes of hash functions. *J. Comput. System Sci.*, 18(2):143–154, 1979.

[15] Uriel Feige and Carsten Lund. On the hardness of computing the permanent of random matrices. *Computational Complexity*, 6(2):101–132, 1996.

[16] Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM J. on Computing*, 22(5):994–1005, October 1993.

[17] Merrick L. Furst, James B. Saxe, and Michael Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17(1):13–27, April 1984.

[18] Rosario Gennaro and Luca Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *41st Annual Symposium on Foundations of Computer Science (Redondo Beach, CA, 2000)*, pages 305–313. IEEE Comput. Soc. Press, Los Alamitos, CA, 2000.

[19] Mikael Goldmann, Mats Näslund, and Alexander Russell. Complexity bounds on general hard-core predicates. *J. Cryptology*, 14(3):177–195, 2001.

[20] Oded Goldreich. *Foundations of Cryptography.* Cambridge University Press, Cambridge, 2001. Basic tools.

[21] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the ACM*, 33(4):792–807, October 1986.

[22] Oded Goldreich, Hugo Krawczyk, and Michael Luby. On the existence of pseudorandom generators. *SIAM J. Comput.*, 22(6):1163–1175, 1993.

[23] Oded Goldreich and Leonid A. Levin. A hard-core predicate for all one-way functions. In ACM [1], pages 25–32.

[24] Oded Goldreich, Noam Nisan, and Avi Wigderson. On Yao's XOR lemma. Technical Report TR95–050, Electronic Colloquium on Computational Complexity, March 1995. `http://www.eccc.uni-trier.de/eccc`.

[25] Johan Håstad. *Computational limitations of small-depth circuits.* MIT Press, 1987.

[26] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM J. Comput.*, 28(4):1364–1396 (electronic), 1999.

[27] Alexander Healy, Salil Vadhan, and Emanuele Viola. Using nondeterminism to amplify hardness. In *Proceedings of the Thirty-Six Annual ACM Symposium on the Theory of Computing*, pages 192–201, Chicago, IL, 13–15 June 2004.

[28] IEEE. *38th Annual Symposium on Foundations of Computer Science*, Miami Beach, Florida, 20–22 October 1997.

[29] Russell Impagliazzo and Moni Naor. Efficient cryptographic schemes provably as secure as subset sum. *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, 9(4):199–216, Fall 1996.

[30] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In ACM [1], pages 44–61.

[31] Michael Kharitonov, Andrew V. Goldberg, and Moti Yung. Lower bounds for pseudorandom number generators. In *30th Annual Symposium on Foundations of Computer Science*, pages 242–247, Research Triangle Park, North Carolina, 30 October–1 November 1989. IEEE.

[32] Leonid A. Levin. One way functions and pseudorandom generators. *Combinatorica*, 7(4):357–363, 1987.

[33] Nathan Linial, Yishay Mansour, and Noam Nisan. Constant depth circuits, Fourier transform, and learnability. *J. Assoc. Comput. Mach.*, 40(3):607–620, 1993.

[34] Richard Lipton. New directions in testing. In *Proceedings of DIMACS Workshop on Distributed Computing and Cryptography*, 1989.

[35] Yishay Mansour, Noam Nisan, and Prasoon Tiwari. The computational complexity of universal hashing. In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing (May 14–16 1990: Baltimore, MD, USA)*, pages 235–243, New York, NY 10036, USA, 1990. ACM Press.

[36] Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology: the journal of the International Association for Cryptologic Research*, 4(2):151–158, 1991.

[37] Moni Naor and Omer Reingold. Number-theoretic constructions of efficient pseudo-random functions. In *38th Annual Symposium on Foundations of Computer Science* [28], pages 458–467.

[38] Noam Nisan. Pseudorandom bits for constant depth circuits. *Combinatorica*, 11(1):63–70, 1991.

[39] Noam Nisan and Avi Wigderson. Hardness vs randomness. *Journal of Computer and System Sciences*, 49(2):149–167, October 1994.

[40] John H. Reif and J. D. Tygar. Efficient parallel pseudo-random number generation. In Hugh C. Williams, editor, *Advances in Cryptology—CRYPTO '85*, volume 218 of *Lecture Notes in Computer Science*, pages 433–446. Springer-Verlag, 1986, 18–22 August 1985.

[41] Omer Reingold, Luca Trevisan, and Salil Vadhan. Notions of reducibility between cryptographic primitives. In *Proceedings of the 1st Theory of Cryptography Conference (Feb 19-21, 2004: Cambridge, MA, USA)*. Springer-Verlag, 2004.

[42] Madhu Sudan, Luca Trevisan, and Salil Vadhan. Pseudorandom generators without the XOR lemma. *J. Comput. System Sci.*, 62(2):236–266, 2001. Special issue on the Fourteenth Annual IEEE Conference on Computational Complexity (Atlanta, GA, 1999).

[43] Luca Trevisan and Salil Vadhan. Pseudorandomness and average-case complexity via uniform reductions. In *Proceedings of the 17th Annual IEEE Conference on Computational Complexity*, pages 129–138, Montréal, CA, May 2002. IEEE.

[44] Emanuele Viola. The complexity of constructing pseudorandom generators from hard functions. Technical Report TR04-020, Electronic Colloquium on Computational Complexity, 2004. `http://www.eccc.uni-trier.de/eccc`. To appear in Computational Complexity. Preliminary version titled 'Hardness vs. Randomness within Alternating Time', in 18th Annual IEEE Conference on Computational Complexity.

[45] Andrew C. Yao. Theory and applications of trapdoor functions (extended abstract). In *23rd Annual Symposium on Foundations of Computer Science*, pages 80–91, Chicago, Illinois, 3–5 November 1982. IEEE.

[46] Jin yi Cai, D. Sivakumar, and Martin Strauss. Constant depth circuits and the lutz hypothesis. In *38th Annual Symposium on Foundations of Computer Science* [28], pages 595–604.

[47] Xiangdong Yu and Moti Yung. Space lower-bounds for pseudorandom-generators. In *Ninth Annual Structure in Complexity Theory Conference*, pages 186–197. IEEE Computer Soc., Los Alamitos, CA, 1994.

A Proof details

Proof of Lemma 3.2. In [44], the following lemma is proved, based on the result by Boppana [11].

Lemma A.1 ([44]). *Let $D : \{0,1\}^n \to \{0,1\}$ be a circuit of size S and depth d. Let $X \in \{0,1\}^n$ be a random input and let $\tilde{X}$ be obtained from X by flipping each bit independently with probability $\delta < 1/2$. Then:*

$$\Pr_{X,\tilde{X}}[D(X) \neq D(\tilde{X})] \leq O(\delta \log^{d-1} S).$$

We obtain Lemma 3.2 as follows:

$$\begin{aligned}
E_{\rho\in R_\delta,U_t,U'_t}\Big[\Delta\Big(C(U_t|_\rho), C(U'_t|_\rho)\Big)\Big] &= \Pr_{\rho\in R_\delta,U_t,U'_t,i}[C(U_t|_\rho)_i \neq C(U'_t|_\rho)_i] \\
&= \Pr_{X,\tilde{X},i}[C(X)_i \neq C(\tilde{X})_i] \\
&\quad \text{(where } \tilde{X} \text{ is obtained from } X \text{ flipping each bit with probability } \delta/2\text{)} \\
&\leq O(\delta \log^{d-1} S). \qquad \text{(by Lemma A.1)}
\end{aligned}$$

□

B Mildly black-box result

In this section we sketch a proof of the fact that (essentially) the negative result in Theorem 1.1, Item (1), holds for *mildly* black-box PRG constructions as well. We do not know if Item (2) holds for mildly black-box PRG constructions.

Mildly black-box constructions are a less restrictive notion of black-box construction than Definition 2.1. We now give the formal definition.

Definition B.1 (Mildly black-box PRG construction). *An oracle machine $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ is a* mildy black-box PRG construction *from one-way function $f : \{0,1\}^n \to \{0,1\}^m$ if for every PPT A there exists an oracle PPT M such that for sufficiently large n, for every $f : \{0,1\}^n \to \{0,1\}^m$ if*

$$\left|\Pr[A(G^f(U_l))] - \Pr[A(U_{l+s})]\right| \geq 1/4$$

then

$$\Pr[f(M^f(f(U_n))) = U_n] \geq 1/n.$$

It seems hard to rule out mildly black-box constructions, since any PRG gives rise to such a construction that simply ignores the oracle. So for mildly black-box constructions one proves that any such construction implies the proof of a 'hard lower-bound'. Intuitively, this means that any such construction is essentially computing the PRG from scratch (i.e., without using the oracle). For more on black-box constructions we refer the reader to the survey in the paper by Reingold, Trevisan and Vadhan [41].

In this section we sketch a proof of the following theorem.

Theorem B.2 (This Paper). *There is a function $\eta(n) \leq o(1)$ such that the following holds: Let $G^f : \{0,1\}^l \to \{0,1\}^{l+s}$ be a mildly black-box PRG construction (Def. B.1) from one-way function in the form in Table* 1.

- *If $s \geq l \cdot \eta(n)$ then $P \neq NP$.*

We now give some intuition about the proof. Consider the proof of our negative result for fully black-box PRG constructions (Section 3). Loosely speaking, we now want the adversary A to be a PPT. One of the problems is that our adversary depends on the function g in Property (II). Intuitively, however, A can invert G^g only knowing the restriction $\rho := \rho_0 \circ h$ (recall g is obtained probabilistically as $g = f_\rho$ for some f). This is because of the low noise sensitivity of constant depth circuits: more formally, we know that the output of the generator $G^{F_\rho}(x)$ will be very biased (over F). Therefore, if A' knew the restriction ρ and the input x to the generator, it could compute $RoundG^g(x)$ which equals to $G^{F_\rho}(x)$ except each bit is rounded according to the bias (over F) of $G^{F_\rho}(x)$. Since G is biased, the output of $RoundG^g(x)$ will be close in Hamming distance to $G^{F_\rho}(x)$. Thus A' could tell if an input comes from the generator by *guessing* an input x and a restriction ρ as follows:

$$A'(z) := 1 \text{ if and only if } \exists x \in \{0,1\}^l, \exists \rho : \Delta(RoundG(x,\rho), z) \leq \eta$$

We need to ensure two things. First A' must be efficient, and second A' should be a good distinguisher. Both problems will be solved by showing a certain pseudorandom distribution on restrictions (for the efficiency of A' we will also use $P = NP$). (Intuitively for A' to be a good distinguisher we need a description of restrictions which is shorter than the stretch of the generator.)

More formally we obtain the following distribution on restrictions:

Lemma B.3. *For every constant c there is a distribution on restrictions $\tilde{R}^c : \{0,1\}^n \to \{0,1,*\}^m$ such that*

1. *There is randomized polynomial time algorithm such that given* $\operatorname{poly}\log n$ *random bits and $i \in \{0,1\}^n$, generates $\rho \in \tilde{R}^c$ and returns $\rho(i)$.*
2. *There is a function $\eta'(n) = o(1)$ such that for every circuit C of size n^c and depth c, and fixed $q_1, \ldots, q_{n^c} \in \{0,1\}^n$: w.p. $1 - \eta'(n)$ over $\rho \in \tilde{R}^c$, $\mathrm{Bias}_{F_\rho}[C(F_\rho(q_1) \ldots F_\rho(q_{n^c}))] \geq 1 - \eta'(n)$, where the bias is only over the choice of F (ρ is fixed).*
3. *The probability over $\rho \in \tilde{R}^c$ that there is $i \in \{0,1\}^n$ such that $\rho(i)$ has less than* $\log^2 n$ *$*$'s is negligible.*

Then, similarly as we did in Section 3, we can prove the following two properties:

i. For every PPT M, with high probability over F and $\rho \in \tilde{R}^c$, M does not invert F_ρ, i.e.:

$$\Pr_{F,U_n}[F_\rho(M(F_\rho(U_n))) = F_\rho(U_n)] \leq \epsilon(n).$$

ii. $E_{F,U_l,\rho\in\tilde{R}^c}\left[\Delta\left(G^{F_\rho}(U_l), \mathit{RoundG}(U_l,\rho)\right)\right] \leq o(1).$

Property (i) can be shown as in Lemma 3.4 using Lemma B.3 (Item (3)). Property (ii) follows from Lemma B.3 (Item (2)). The rest of the proof is similar to the proof on Page 8: A counting argument shows that A' is a good distinguisher when the stretch of the PRG is too big (i.e. $s \geq l \cdot \eta(n)$). Here we use Property (ii). Then one shows that A' is efficient (i.e. a PPT) under the assumption that $P = NP$ (because if $P = NP$ then guessing x, ρ and computing RoundG can be done in polynomial time). So by definition of mildly black-box PRG construction there is a PPT M that inverts F_ρ. But this contradicts Property (i).

We conclude this section giving some intuition of how to obtain the distribution on restrictions in Lemma B.3. Recall our distribution on restrictions in Section 3 consisted of a hash function $h : \{0,1\}^n \to [b]$, for $b = n^{\log n}$, and a restriction $\rho_0 : [b] \to \{0,1,*\}^m$. We obtain the distribution in Lemma B.3 by derandomizing both h and ρ_0.

Derandomization of h: For h we use any family of hash functions which can be generated using poly $log(n)$ random bits and that has low collision probability, for example the following standard construction (which we use with $m := \log n$.)

Lemma B.4. *Let n be a prime power. For every m there is a family of hash functions $\tilde{h}_s : \{0,1\}^n \to \{0,1\}^{m\log n}$ with seed length $|s| = m\log n$ such that for every $x \neq y$, $\Pr_s[\tilde{h}_s(x) = \tilde{h}_s(y)] \leq 1/n^{m-1}$.*

Proof. Construction Fix a field F of size n^m. The seed s represents an element in F. We see x as a univariate polynomial $p_x : F \to F$ where p_x has degree $n/m \log n \leq n$. Then $\tilde{h}_s(x) := p_x(s)$.

Analysis: Fix $x \neq y$. Then clearly $p_x \neq p_y$. So

$$\Pr_s[p_x(s) = p_y(s)] \leq n/F = 1/n^{m-1}$$

where we use the well known fact that two distinct polynomial of degree $\leq n$ over F agree in at most n/F fraction of points. □

Derandomization of ρ_0: The derandomization of ρ_0 is similar to what we did in Section 6.1. It again uses Nisan's *unconditional* PRG against constant depth circuits [38]. More formally, Nisan shows a PRG $Nis : \{0,1\}^{\mathrm{poly}\log n} \to \{0,1\}^{\mathrm{poly}(n)}$ that is pseudorandom for constant depth circuits of size poly(n) (where the exponent in the poly log depends on the depth of the circuit). That is, for every constant depth circuit $C : \{0,1\}^{\mathrm{poly}(n)} \to \{0,1\}$ of size poly(n):

$$\left|\Pr[C(Nis(U_{\mathrm{poly}\log n})) = 1] - \Pr[C(U_{\mathrm{poly}(n)})]\right| = o(1).$$

Our new distribution on restrictions $\rho_0 : [b] \to \{0,1,*\}^m$ is then obtained by plugging a random seed into Nisan's PRG and interpreting its output bits as choices for $\{0,1,*\}$. Call $\tilde{\rho_0}$ this pseudorandom distribution on restrictions.

We must ensure that Items (2) and (3) in Lemma B.3 hold.

Item (2): We need to ensure that w.h.p. for every $i \in \{0,1\}^n$ $\tilde{\rho_0}(i)$ has $\log^2 n$ *'s . Since this holds for a truly random restriction $\rho_0 : [b] \to \{0,1,*\}^m$ (by a concentration bound), it would be enough to show that we can check with a constant depth circuit if a block has $\log^2 n$ *'s. Then the result follows from pseudorandomness of Nisan's generator *Nis* . But this seems problematic, because it is known that constant depth circuits cannot count! (See e.g. [25].) However, it was shown by Ajtai and Ben-Or that constant depth circuits of size poly(n) *can* count up to $\log^2 n$ [4]. So using their result we can guarantee that $\tilde{\rho_0}$ is such that w.h.p. for every $i \in [b]$, $\tilde{\rho_0}(i)$ has at least $\log^2 n$ *'s.

Item (3): First we argue that under $\tilde{\rho_0}$ constant depth circuits still have high bias (over the choice of the random bits for the *'s). This is obtained by showing (as in Section 6.1) that the bias of a constant depth circuit C is essentially the acceptance probability of another (slightly bigger) constant depth circuit C', then arguing by pseudorandomness of N (details omitted). To obtain Item (3) we argue as in the proof of Property (II) in Section 3. By an appropriate choice of parameters for the hash function in Lemma B.4 (i.e. $m := \log n$) we can ensure that for every fixed $q_1, \ldots, q_r$, their images through $\tilde{h}$ will be pairwise different w.p. $1 - \epsilon(n)$. Whenever this happens we have that the circuit will have high bias because what we said above about $\tilde{\rho_0}$.